**第02章 查询和排序**

**本章内容**

* Scott用户案例下表分析
* 常见操作
* distinct去掉重复行
* 使用where条件过滤行
* 使用order by排序

scott、system、sys

# Scott用户案例下表分析

## 雇员表：EMP



## 部门表：Dept



## 工资等级表：Salgrade



## 奖金表：Bonus



查看表结构命令：desc 表名

# 复制scott用户下所有的表数据

注意：把scott\_data.sql文件中所有的“LANQIAO”替换成登录的账号名称

例：如果是“zs”登录，则把“LANQIAO”字样替换成“zs”即可.

-----------------------------------------------------------

-- Export file for user LANQIAO --

-- Created by Administrator on 2015/4/19 星期日, 下午 3:02:24 --

-----------------------------------------------------------

spool LANQIAO用户下所有表结构.log

prompt

prompt Creating table BONUS

prompt ====================

prompt

create table LANQIAO.BONUS

(

ename VARCHAR2(10),

job VARCHAR2(9),

sal NUMBER,

comm NUMBER

)

tablespace USERS

pctfree 10

initrans 1

maxtrans 255;

prompt

prompt Creating table DEPT

prompt ===================

prompt

create table LANQIAO.DEPT

(

deptno NUMBER(2) not null,

dname VARCHAR2(14),

loc VARCHAR2(13)

)

tablespace USERS

pctfree 10

initrans 1

maxtrans 255

storage

(

initial 64K

next 1M

minextents 1

maxextents unlimited

);

alter table LANQIAO.DEPT

add constraint PK\_DEPT primary key (DEPTNO)

using index

tablespace USERS

pctfree 10

initrans 2

maxtrans 255

storage

(

initial 64K

next 1M

minextents 1

maxextents unlimited

);

prompt

prompt Creating table EMP

prompt ==================

prompt

create table LANQIAO.EMP

(

empno NUMBER(4) not null,

ename VARCHAR2(10),

job VARCHAR2(9),

mgr NUMBER(4),

hiredate DATE,

sal NUMBER(7,2),

comm NUMBER(7,2),

deptno NUMBER(2)

)

tablespace USERS

pctfree 10

initrans 1

maxtrans 255

storage

(

initial 64K

next 1M

minextents 1

maxextents unlimited

);

alter table LANQIAO.EMP

add constraint PK\_EMP primary key (EMPNO)

using index

tablespace USERS

pctfree 10

initrans 2

maxtrans 255

storage

(

initial 64K

next 1M

minextents 1

maxextents unlimited

);

alter table LANQIAO.EMP

add constraint FK\_DEPTNO foreign key (DEPTNO)

references LANQIAO.DEPT (DEPTNO);

prompt

prompt Creating table SALGRADE

prompt =======================

prompt

create table LANQIAO.SALGRADE

(

grade NUMBER,

losal NUMBER,

hisal NUMBER

)

tablespace USERS

pctfree 10

initrans 1

maxtrans 255

storage

(

initial 64K

next 1M

minextents 1

maxextents unlimited

);

spool off

prompt PL/SQL Developer import file

prompt Created on 2015年4月19日 星期日 by Administrator

set feedback off

set define off

prompt Disabling triggers for BONUS...

alter table BONUS disable all triggers;

prompt Disabling triggers for DEPT...

alter table DEPT disable all triggers;

prompt Disabling triggers for EMP...

alter table EMP disable all triggers;

prompt Disabling triggers for SALGRADE...

alter table SALGRADE disable all triggers;

prompt Disabling foreign key constraints for EMP...

alter table EMP disable constraint FK\_DEPTNO;

prompt Deleting SALGRADE...

delete from SALGRADE;

commit;

prompt Deleting EMP...

delete from EMP;

commit;

prompt Deleting DEPT...

delete from DEPT;

commit;

prompt Deleting BONUS...

delete from BONUS;

commit;

prompt Loading BONUS...

prompt Table is empty

prompt Loading DEPT...

insert into DEPT (deptno, dname, loc)

values (10, 'ACCOUNTING', 'NEW YORK');

insert into DEPT (deptno, dname, loc)

values (20, 'RESEARCH', 'DALLAS');

insert into DEPT (deptno, dname, loc)

values (30, 'SALES', 'CHICAGO');

insert into DEPT (deptno, dname, loc)

values (40, 'OPERATIONS', 'BOSTON');

commit;

prompt 4 records loaded

prompt Loading EMP...

insert into EMP (empno, ename, job, mgr, hiredate, sal, comm, deptno)

values (7369, 'SMITH', 'CLERK', 7902, to\_date('17-12-1980', 'dd-mm-yyyy'), 800, null, 20);

insert into EMP (empno, ename, job, mgr, hiredate, sal, comm, deptno)

values (7499, 'ALLEN', 'SALESMAN', 7698, to\_date('20-02-1981', 'dd-mm-yyyy'), 1600, 300, 30);

insert into EMP (empno, ename, job, mgr, hiredate, sal, comm, deptno)

values (7521, 'WARD', 'SALESMAN', 7698, to\_date('22-02-1981', 'dd-mm-yyyy'), 1250, 500, 30);

insert into EMP (empno, ename, job, mgr, hiredate, sal, comm, deptno)

values (7566, 'JONES', 'MANAGER', 7839, to\_date('02-04-1981', 'dd-mm-yyyy'), 2975, null, 20);

insert into EMP (empno, ename, job, mgr, hiredate, sal, comm, deptno)

values (7654, 'MARTIN', 'SALESMAN', 7698, to\_date('28-09-1981', 'dd-mm-yyyy'), 1250, 1400, 30);

insert into EMP (empno, ename, job, mgr, hiredate, sal, comm, deptno)

values (7698, 'BLAKE', 'MANAGER', 7839, to\_date('01-05-1981', 'dd-mm-yyyy'), 2850, null, 30);

insert into EMP (empno, ename, job, mgr, hiredate, sal, comm, deptno)

values (7782, 'CLARK', 'MANAGER', 7839, to\_date('09-06-1981', 'dd-mm-yyyy'), 2450, null, 10);

insert into EMP (empno, ename, job, mgr, hiredate, sal, comm, deptno)

values (7788, 'SCOTT', 'ANALYST', 7566, to\_date('19-04-1987', 'dd-mm-yyyy'), 3000, null, 20);

insert into EMP (empno, ename, job, mgr, hiredate, sal, comm, deptno)

values (7839, 'KING', 'PRESIDENT', null, to\_date('17-11-1981', 'dd-mm-yyyy'), 5000, null, 10);

insert into EMP (empno, ename, job, mgr, hiredate, sal, comm, deptno)

values (7844, 'TURNER', 'SALESMAN', 7698, to\_date('08-09-1981', 'dd-mm-yyyy'), 1500, 0, 30);

insert into EMP (empno, ename, job, mgr, hiredate, sal, comm, deptno)

values (7876, 'ADAMS', 'CLERK', 7788, to\_date('23-05-1987', 'dd-mm-yyyy'), 1100, null, 20);

insert into EMP (empno, ename, job, mgr, hiredate, sal, comm, deptno)

values (7900, 'JAMES', 'CLERK', 7698, to\_date('03-12-1981', 'dd-mm-yyyy'), 950, null, 30);

insert into EMP (empno, ename, job, mgr, hiredate, sal, comm, deptno)

values (7902, 'FORD', 'ANALYST', 7566, to\_date('03-12-1981', 'dd-mm-yyyy'), 3000, null, 20);

insert into EMP (empno, ename, job, mgr, hiredate, sal, comm, deptno)

values (7934, 'MILLER', 'CLERK', 7782, to\_date('23-01-1982', 'dd-mm-yyyy'), 1300, null, 10);

commit;

prompt 14 records loaded

prompt Loading SALGRADE...

insert into SALGRADE (grade, losal, hisal)

values (1, 700, 1200);

insert into SALGRADE (grade, losal, hisal)

values (2, 1201, 1400);

insert into SALGRADE (grade, losal, hisal)

values (3, 1401, 2000);

insert into SALGRADE (grade, losal, hisal)

values (4, 2001, 3000);

insert into SALGRADE (grade, losal, hisal)

values (5, 3001, 9999);

commit;

prompt 5 records loaded

prompt Enabling foreign key constraints for EMP...

alter table EMP enable constraint FK\_DEPTNO;

prompt Enabling triggers for BONUS...

alter table BONUS enable all triggers;

prompt Enabling triggers for DEPT...

alter table DEPT enable all triggers;

prompt Enabling triggers for EMP...

alter table EMP enable all triggers;

prompt Enabling triggers for SALGRADE...

alter table SALGRADE enable all triggers;

set feedback on

set define on

prompt Done.

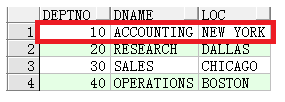
# 常见操作

## 如何查看表的数据

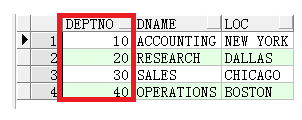
使用PL/SQL Developer查看表的数据, 展开左边导航树的Tables节点,选中想要查看数据的dept表，右键，选择”query data”即可

## 记录和字段

记录:表中的一行数据叫做一条记录



字段:表中的一列叫做一个字段



## 查询语句

select 字段 from 表名 where 条件 order by 排序

或

select 字段

from 表名

[where 条件]

[order by 排序]

## 查询emp表

例1: 查询emp表全部数据

select \* from emp;

例2: 查询指定字段(过滤列)编号及姓名

select empno,ename from emp;

例3: 查询员工的姓名和年薪

select ename,sal\*12 from emp;

## 虚表(dual)

select 3\*4 from dual

## 给字段加别名（？）

查询员工姓名和年薪，年薪字段显示成 "年薪"，语句如下

方法一：加as

select ename,sal\*12 as 年薪 from emp

方法二：不加as

select ename,sal\*12 "年薪" from emp

## 字符串连接

Java中字符串连接是 + 号 ,oracle中字符串连接是 || 号

例：查员工的姓名和月薪, 要求数据的显示格式如下

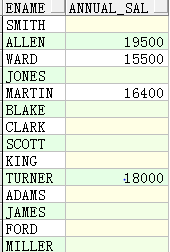
姓名: SMITH，月薪:800

select '姓名:'||ename || ',月薪:' || sal from emp

## NVL函数

求真正的年薪(带年终奖),使用如下语句会发生错误,如下左图

select ename,sal\*12+comm as annual\_sal from emp



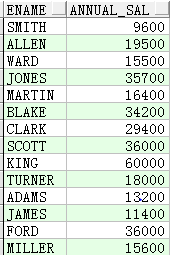
原因是null和任何数参与四则运算都返回null

nvl(comm,0) 如果第一个参数为null，则返回第二个参数(0)，

如果第一个参数不为null,则函数返回第一个参数本身

select ename,sal\*12+nvl(comm,0) as annual\_sal from emp

如图



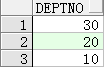
## 查询系统当前时间

select sysdate from dual

# distinct去掉重复行

例：查看员工到底来自哪几个部门，sql语句如下

select distinct deptno from emp



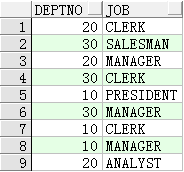
例：查一下一共有几个工种,sql语句如下

select distinct job from emp



例：如果distinct修饰两个字段，代表两个字段组合起来不能重复

select distinct deptno,job from emp



# 数据查询

## 简单查询

例1、查询dept表全部数据

select \* from dept;

## 条件查询

符号：>、<、>=、<=、<>、in、between......and、is null

语法：select \* from 表名 where 条件

例：查询部门编号为10的员工的全部信息

select \* from emp where deptno=10;

例:查询薪水大于1500的员工姓名，工资

select ename,sal from emp where sal>1500;

例:查询部门编号为10并且薪水大于1000的员工的姓名和工资

select ename,sal from emp where deptno=10 and sal>1000;

例:查询部门编号不是10的员工姓名和薪水以及部门编号

select ename,sal ,deptno from emp where deptno<>10;

select ename,sal ,deptno from emp where deptno!=10;

注意:<>表示不等于,也可以用!=表示

例:查询没有奖金的员工姓名和工资、奖金

select ename，sal，comm from emp where comm is null;

## 逻辑查询

符号：and、or、not

例:查询薪水大于等于800小于等于1500的员工姓名和工资

select ename，sal from emp where sal >= 800 and sal<= 1500;

select ename，sal from emp where sal between 800 and 1500;

例：查询薪水是800或1500或2000的员工的姓名和薪水、奖金

select ename,sal,comm from emp where sal=800 or sal=1500 or sal=2000;

select ename,sal,comm from emp where sal in(800,1500,2000);

## 模糊查询

使用like关键字，支持条件的模糊查询

\_ :占一个字母

% :代表占0个或多个字母

其余的字符代表本身

要想查带%的字符串 需要用到转义字符\%

例:查询姓名里面包含ALL的员工姓名

select ename from emp where ename like '%ALL%'

例:查询第二个字母为A的员工姓名

select ename from emp where ename like '\_A%'

# 排序

使用order by对过滤后的数据进行排序 asc表示升序 desc表示降序, 注意:不写asc的话 默认按升序排列

语法：select \* from 表名 order by 字段 desc|asc

例：查询所有部门信息，按部门编号降序排列

select \* from dept order by deptno desc

例：查询所有的员工的信息，根据薪水升序排列

select \* from emp order by sal asc

例：查询员工的姓名、薪水、部门编号，先按部门编号升序、部门编号相同的按姓名降序

select ename,sal,deptno from emp order by deptno asc,ename desc